V3: Secondary Aims Analysis

Using moderators and Q-learning to construct more deeply-tailored Adaptive Interventions

Mason Ferlic

Jamie Yap

John J. Dziak

Daniel Almirall

# Learning Goals

Typical secondary aims analysis involves using data from a SMART to learn a set of decision rules for a more deeply-tailored adaptive intervention that leads to better outcomes. In this workbook, we will use a subset of candidate moderators to test which variables are useful for tailoring first and second-stage intervention options in a 2-stage SMART.

To learn an optimal decision rule in a 2-arm RCT we could simply fit a moderated regression to determine if baseline covariates are useful for tailoring treatment assignment. In a prototypical SMART we have two stages of treatment assignment and you may be thinking, can’t we just fit moderated regressions for first and second-stage, and taken together, we have a proposal for a more deeply-tailored AI? The answer is no! These separate regressions don’t “talk” to each other. That is, we want to make the optimal first-stage decision accounting for the fact that, in the future, we will be making an optimal second-stage decision.

In this workbook we will:

1. Learn how moderators of first and second-stage treatment effects can be used to construct optimal adaptive interventions
   * Fit moderated regression models
   * Test contrasts and generate interaction plots
2. Learn how Q-learning is used to estimate the effect of a more-deeply tailored AI.
   * Implement using the R Package qlaci

# Setup

Load required packages

library(geepack)  
library(ggplot2)  
library(dplyr)  
library(emmeans)  
library(qlaci)

# Load data

This is data that was *simulated* to mimic data arising from the ADHD SMART study (PI: William Pelham). An accompanying handout (“ADHD\_SMART\_handout.pdf”) describes the variables in the data set.

# Load data  
dat\_adhd <- read.csv("data/adhd-simulated-2023.csv")

|  |
| --- |
| ADHD study prototypical SMART |

## Examine data

|  |
| --- |
| **ADHD dataset variable descriptions**  **Baseline covariates:**   * ID subject identifier * odd Oppositional Defiant Disorder diagnosis, reflecting whether the child was (coded as 1) or was not (coded as 0) diagnosed with ODD before the first-stage intervention. * severity ADHD score, reflecting ADHD symptoms at the end of the previous school year (larger values reflect greater symptoms). Range 0-10. * priormed medication prior to first-stage intervention, reflecting whether the child did (coded as 1) or did not (coded as 0) receive medication during the previous school year. * race white (coded 1) versus non-white (coded 0).   **Intermediate covariates:**   * R response status. R = 0 if child was classified as non-responder to first stage intervention, R= 1 if they were classified as a responder. * NRtime months at which child was classified as non-responder. Range 2-8. Undefined for responders. * adherence adherence to the stage 1 intervention. Reflecting whether the child did (coded as 1) or did not (coded as 0) show high adherence to initial treatment.   **Treatments:**  We use effect coding (average to zero) to denote the two levels of treatment assignment. The primary benefit of effect coding is that we get interpretable estimates of both the main effects and interactions.   * A1 stage 1 treatment assignment. Randomized with probability to Medication (MED, ) or Behavioral Intervention (BMOD, ). * A2 stage 2 treatment assignment for non-responders. Non-responders we randomized with probability to receive Augmented (AUG, ) or Intensified (INT, ) care. Undefined for responders.   **Outcomes**   * Y0 baseline school performance (higher values reflect better performance). * Y1 mid-year school performance. * Y2 end-of-year school performance (primary outcome for analysis) |

head(dat\_adhd, n = 10)

ID odd severity priormed race Y0 A1 R NRtime adherence Y1 A2 Y2  
1 1 1 2.88 0 1 2.32 -1 0 4 0 2.79 1 0.598  
2 2 0 4.13 0 0 2.07 1 1 NA 1 2.20 NA 4.267  
3 3 1 5.57 0 1 1.00 -1 1 NA 0 2.29 NA 1.454  
4 4 0 4.93 0 1 3.23 1 0 4 0 3.05 -1 6.762  
5 5 1 5.50 0 1 1.48 1 0 6 0 1.73 -1 3.580  
6 6 0 5.50 0 1 1.72 1 0 3 0 2.40 1 2.075  
7 7 0 6.79 0 1 2.26 1 0 7 0 2.84 1 2.594  
8 8 0 4.32 0 1 2.81 1 1 NA 0 2.75 NA 4.051  
9 9 1 9.09 1 1 1.90 1 0 6 0 3.59 -1 2.970  
10 10 0 6.09 0 1 2.50 1 0 5 1 2.55 1 6.022

## Clean/transform data

Here we grand mean center the baseline covariates. This does not change the point estimates, but is useful when interpreting model coefficients or when hand coding contrasts.

# Grand mean center all baseline covariates, append '\_c' for centered  
dat\_adhd\_c <- dat\_adhd %>% mutate(odd\_c = odd - mean(odd),  
 severity\_c = severity - mean(severity),  
 priormed\_c = priormed - mean(priormed),  
 race\_c = race - mean(race))

# Q-learning

We use Q-learning to learn a set of decision rules for a more deeply-tailored adaptive intervention which combines optimal tactics at each stage.

Q-learning has three steps:

1. Fit a stage 2 moderated regression
2. Predict the stage 2 outcome under the optimal decision rule:
3. Fit a stage 1 moderated regression on

## Step 1: moderators of stage 2 treatment

A common secondary aim is to learn whether we can use baseline or intermediate covariates to select an optimal second-stage tactic for **non-responders.** For example, whether covariates such as the initial treatment and adherence to initial treatment adherence moderate the effect of .

We hypothesize that for children who are non-adherent to first-stage treatment it will be better to AUGment () with the alternative treatment as opposed to INTensifying () the current treatment.

### Regression model

We test this secondary aim by fitting a **moderated regression model** using data from non-responders. This model examines whether binary intermediate outcome variable adherence, and first-stage treatment , moderate the effect of second-stage treatment on end-of-year outcome , controlling for other baseline covariates . The model is as follows:

|  |
| --- |
| Note |
| We interact with the uncentered variable adherence since we are interested in the effect of at the each of the two levels: adherent (0) and non-adherent (1). |

First, subset data.frame to non-responders

dat\_adhd\_nr <- subset(dat\_adhd, R == 0) # subset data.frame to non-responders  
  
head(dat\_adhd\_nr) # view top 6 rows

ID odd severity priormed race Y0 A1 R NRtime adherence Y1 A2 Y2  
1 1 1 2.88 0 1 2.32 -1 0 4 0 2.79 1 0.598  
4 4 0 4.93 0 1 3.23 1 0 4 0 3.05 -1 6.762  
5 5 1 5.50 0 1 1.48 1 0 6 0 1.73 -1 3.580  
6 6 0 5.50 0 1 1.72 1 0 3 0 2.40 1 2.075  
7 7 0 6.79 0 1 2.26 1 0 7 0 2.84 1 2.594  
9 9 1 9.09 1 1 1.90 1 0 6 0 3.59 -1 2.970

Next, fit a moderated regression to the subset of non-responders. Interact with and adherence

# geeglm fits a Generalize Estimating Equation. Similar to lm(), but gives us robust standard errors.  
mod2 <- geepack::geeglm(Y2 ~ odd + severity + priormed + race + adherence + A1 + A2 + A1:A2 + adherence:A2,   
 id = ID, # a subject identifier is required for geeglm()  
 data = dat\_adhd\_nr)  
  
summary(mod2)

Call:  
geepack::geeglm(formula = Y2 ~ odd + severity + priormed + race +   
 adherence + A1 + A2 + A1:A2 + adherence:A2, data = dat\_adhd\_nr,   
 id = ID)  
  
 Coefficients:  
 Estimate Std.err Wald Pr(>|W|)   
(Intercept) 2.7939 0.4905 32.44 1.2e-08 \*\*\*  
odd -0.6239 0.2748 5.16 0.02316 \*   
severity -0.0496 0.0684 0.53 0.46821   
priormed -0.5773 0.3348 2.97 0.08464 .   
race 0.2627 0.4170 0.40 0.52873   
adherence 0.9272 0.2561 13.11 0.00029 \*\*\*  
A1 0.4249 0.1518 7.83 0.00513 \*\*   
A2 -1.1725 0.1675 49.01 2.6e-12 \*\*\*  
A1:A2 -0.1523 0.1245 1.50 0.22120   
adherence:A2 1.7331 0.2553 46.08 1.1e-11 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Correlation structure = independence   
Estimated Scale Parameters:  
  
 Estimate Std.err  
(Intercept) 1.6 0.203  
Number of clusters: 101 Maximum cluster size: 1

|  |
| --- |
| Warning |
| Only the second-stage coefficients from the moderated regression in [Equation 1](#eq-mod2) are causal. The first-stage coefficients are biased as a result of sub-setting to non-responders. |

The regression model from [Equation 1](#eq-mod2) gives us the effect of among non-responders. Looking at the output we find that adherence is a significant moderator of stage 2 treatment, but is not.

### Knowledge check #1

|  |
| --- |
| Questions |
| 1. Why do we subset to non-responders when fitting a second-stage moderated regression? 2. What does this regression tell us about the use of as a tailoring variable? |

### Marginal means of stage 2

We will use a very powerful package called emmeans to estimate the marginal mean of end-of-year school performance under different treatment/covariate options. We could do this by writing custom contrasts of the model coefficients, but emmeans does this for us.

|  |
| --- |
| Note |
| Marginal means refers to estimating the expected value while holding some covariates constant and averaging over the rest. |

We use the fitted moderated regression model given in [Equation 1](#eq-mod2) to estimate the average effect of second-stage treatment, among non-responders, conditional on levels of adherence.

# The formula notation denotes the mean outcome under all combination of the factors A1, A2, given levels of adherence.  
# We specify weights=proportional since we want to average over the observed distribution of the other baseline covariates.  
# df of Inf indicates CIs are computed using the standard normal distribution  
  
em2 <- emmeans::emmeans(mod2, ~ A1 + A2 | adherence, weights = "proportional")  
print(em2)

adherence = 0:  
 A1 A2 emmean SE df asymp.LCL asymp.UCL  
 -1 -1 2.96 0.373 Inf 2.22 3.69  
 1 -1 4.11 0.282 Inf 3.56 4.66  
 -1 1 0.92 0.256 Inf 0.41 1.42  
 1 1 1.46 0.305 Inf 0.86 2.06  
  
adherence = 1:  
 A1 A2 emmean SE df asymp.LCL asymp.UCL  
 -1 -1 2.15 0.383 Inf 1.40 2.90  
 1 -1 3.30 0.353 Inf 2.61 4.00  
 -1 1 3.58 0.327 Inf 2.93 4.22  
 1 1 4.12 0.267 Inf 3.60 4.64  
  
Results are averaged over the levels of: odd, priormed, race   
Covariance estimate used: vbeta   
Confidence level used: 0.95

The results from the moderated regression and the marginal means output suggest that for those children who are non-adherent (adherence = 0) it is better to Augment () rather than intensify. Likewise, for those children that are adherent (adherence = 1) it is better to Intensify (). Notice, our optimal tactic does not depend on first-stage treatment .

### Interaction plot of stage 2

We can also use the emmeans package to visualize the estimated mean outcome for non-responders under each tactic.

ep2 <- emmeans::emmip(em2, A1\*A2 ~ adherence, style = "factor") # Interaction plot for trace.factors ~ x.factors

# Prettify the plot  
ep2$data %>% mutate(across(1:3, as.factor)) %>%  
 ggplot(aes(xvar, yvar, color = A2, linetype = A1, group = tvar)) +  
 geom\_line() +  
 geom\_point() +  
 scale\_color\_manual("A2", values = c("-1" = "darkgreen", "1" = "purple"),   
 labels = c("-1" = "-1 AUG", "1" = "1 INT")) +  
 scale\_linetype\_manual("A1", values = c("-1" = 1,"1" = 6),   
 labels = c("-1" = "-1 MED", "1" = "1 BMOD")) +  
 labs(title = "Moderator analysis of stage 2 intervention options",  
 x = "Adherence to stage 1",  
 y = "EOS School Performance \n (higher is better)") +  
 scale\_x\_discrete(labels = c("Non-adherent", "Adherent")) +  
 theme\_classic()

|  |
| --- |
|  |

### Knowledge check #2

|  |
| --- |
| Questions |
| 1. What is the optimal tactic for non-adhering, non-responders to BMOD(1)? To MED(-1)? 2. was found to be not useful for tailoring, should we still include it in the regression model? |

## Step 2: predict optimal outcome

The optimal second-stage tactic learned from the moderated regression:

Create new data.frame with optimal for every non-responding child

# Assign optimal second-stage tactic using learned decision rule  
dat\_adhd\_nr\_opt2 <- dat\_adhd\_nr %>%   
 mutate(A2 = if\_else(adherence == 1, 1, -1))

Predicted outcome for non-responders under the optimal treatment assignment

dat\_adhd\_nr\_opt2$Y2\_opt <- predict(mod2, newdata = dat\_adhd\_nr\_opt2) # using the stage 2 moderated regression model

Merge non-responders with the observed outcome from responders

# Responders get assigned their observed outcome (no stage 2 tactic)  
dat\_adhd\_r <- dat\_adhd %>% filter(R == 1) %>%  
 mutate(Y2\_opt = Y2)  
  
# combine non-responders w/ responders  
dat\_adhd\_opt2 <- bind\_rows(dat\_adhd\_nr\_opt2, dat\_adhd\_r)

We now have a data frame with the estimated optimal outcome for non-responders tailored by adherence.

## Step 3: first-stage tailoring controlling for optimal second-stage

A common secondary aim is whether we can now use baseline information to learn an optimal first-stage tactic to build a more deeply-tailored AI.

We hypothesize that children already on medication (priormed = 1) will be better-off, on average, starting with MED () instead of BMOD () due to parent/child habituation with taking medication.

### Regression model

We fit a moderated regression model for first-stage treatment using the dat\_adhd\_opt2 data frame, which accounts for the optimal future second-stage tactic. The tailoring variable of interest is priormed.

# Moderator regression for first stage tailoring on priormed, controlling for optimal future tactic  
Qmod1 <- geepack::geeglm(Y2\_opt ~ odd + severity + race + A1\*priormed,  
 id = ID,  
 data = dat\_adhd\_opt2)  
  
summary(Qmod1)

Call:  
geepack::geeglm(formula = Y2\_opt ~ odd + severity + race + A1 \*   
 priormed, data = dat\_adhd\_opt2, id = ID)  
  
 Coefficients:  
 Estimate Std.err Wald Pr(>|W|)   
(Intercept) 3.7927 0.2766 187.99 < 2e-16 \*\*\*  
odd -0.5735 0.1342 18.27 1.9e-05 \*\*\*  
severity -0.1161 0.0364 10.18 0.00142 \*\*   
race 0.7274 0.1896 14.72 0.00012 \*\*\*  
A1 0.6947 0.0709 96.00 < 2e-16 \*\*\*  
priormed -0.0362 0.1710 0.04 0.83245   
A1:priormed -0.9072 0.1707 28.25 1.1e-07 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Correlation structure = independence   
Estimated Scale Parameters:  
  
 Estimate Std.err  
(Intercept) 0.644 0.106  
Number of clusters: 150 Maximum cluster size: 1

|  |
| --- |
| Warning |
| The standard errors here are potentially incorrect because they don’t take into account sampling error in estimation of . Luckily, we provide software (see qlaci) to do proper inference! |

We use emmeans package to estimate the expected end-of-year school performance for an adaptive intervention that offers BMOD(1) or MED(-1) at first-stage for levels of priormed, adjusting for the fact we are optimally tailoring second-stage treatment for non-responders by adherence.

qem1 <- emmeans::emmeans(Qmod1, ~ A1 | priormed, weights = "proportional")  
summary(qem1, infer = FALSE)[, 1:3]

A1 priormed emmean  
1 -1 0 2.93  
2 1 0 4.32  
3 -1 1 3.80  
4 1 1 3.37

### Interaction plot of optimal tactic

# Interaction plot of A1 with priormed  
qep1 <- emmeans::emmip(Qmod1, A1 ~ priormed, style = "factor", weights = "proportional")

# Prettify plot   
qep1$data %>% mutate(across(1:2, as.factor)) %>%  
 ggplot(aes(xvar, yvar, color = A1, group = tvar)) +  
 geom\_line(linewidth = 1) +  
 geom\_point() +  
 scale\_color\_manual("A1", values = c("-1" = "red", "1" = "blue"),   
 labels = c("-1" = "-1 MED", "1" = "1 BMOD")) +  
 theme\_classic() +  
 labs(title = "Moderator of stage 1 controlling for optimal stage 2",  
 x = "Medication use in Prior year",  
 y = "EOS School Performance \n (higher is better)") +  
 scale\_x\_discrete(labels = c("No prior med", "Prior med")) +  
 scale\_y\_continuous(n.breaks = 8)

|  |
| --- |
|  |

### Estimated mean under the more deeply-tailored AI

The results of Q-learning generated a proposal for a more deeply-tailored AI that tailors first-stage on priormed and second-stage on response status and adherence:

|  |
| --- |
| The more deeply-tailored adaptive intervention learned using Q-learning |

To estimate the mean outcome under the proposed decision rules we essentially repeat step 2 and create a new data frame with the optimal first-stage tactic by levels of priormed. We use the parametric g-formula and the fitted model from [Equation 2](#eq-Qmod1) to compute the mean outcome if every child had received the more deeply-tailored adaptive intervention.

# Assign optimal second-stage tactic  
dat\_adhd\_opt1 <- dat\_adhd\_opt2 %>%   
 mutate(A1 = if\_else(priormed == 1, -1, 1)) # learned decision rule  
  
# Predict outcome under optimal treatment a2  
dat\_adhd\_opt1$Y2\_opt <- predict(Qmod1, newdata = dat\_adhd\_opt1)  
  
cat("Estimated school performance under optimal AI:", mean(dat\_adhd\_opt1$Y2\_opt))

Estimated school performance under optimal AI: 4.16

### Knowledge check #3

|  |
| --- |
| Questions |
| 1. Why do we need to use Q-learning to estimate a more deeply-tailored AI? What is wrong with simply running two moderator regressions? 2. In the regression model Qmod1 what does the main effect of represent? 3. If priormed did not moderate the effect of would this still be a valid more deeply-tailored AI? |

The steps we just covered illustrate the basics of Q-learning, but we only get point estimates, not standard errors. For that we need some extra software… enter qlaci()!

# The qlaci package

The qlaci package performs Q-learning on data arising from a two-stage SMART. It is useful when we need standard errors for the estimates of our more deeply-tailored adaptive intervention.

|  |
| --- |
| Note |
| The qlaci package can be downloaded from the d3c github: [d3center-isr/qlaci](https://github.com/d3center-isr/qlaci) |

First, we specify the contrast matrix that will be used for the stage 1 regression (step 3 of Q-learning). qlaci() uses this matrix to estimate the mean outcomes under each of the first-stage treatments (accounting for the future optimal decision) at both levels of priormed. We also specify a contrast that estimates the mean outcome if everyone in the study had received the optimal more deeply-tailored AI.

prob.pm <- mean(dat\_adhd\_c$priormed) # probability of priormed, used to calculate mean under optimal decision rules  
  
## contrast matrix - we must transpose this for qlaci  
c1 <-  
 rbind(  
 "Mean Y under bmod, prior med" = c(1, rep(0, 3), 1, 1, 1),  
 "Mean Y under med, prior med" = c(1, rep(0, 3), 1, -1, -1),  
 "Mean diff (bmod-med) for prior med" = c(0, rep(0, 3), 0, 2, 2),  
 "Mean Y under bmod, no prior med" = c(1, rep(0, 3), 0, 1, 0),  
 "Mean Y under med, no prior med" = c(1, rep(0, 3), 0, -1, 0),  
 "Mean diff (bmod-med) for no prior med" = c(0, rep(0, 3), 0, 2, 0),  
 "Mean Y Optimal AI" = c(1, rep(0, 3), prob.pm, 1 - 2 \* prob.pm, -prob.pm)  
 )

|  |
| --- |
| **qlaci() parameters**  The following are the arguments we need to provide to qlaci():   * H10: Baseline covariates we want to adjust for in the first-stage regression. * H11: Variables that interact with first-stage treatment in the first-stage regression (candidate variables for deeper tailoring). * A1: Indicator for first-stage treatment * Y1: A continuous intermediate outcome. Here, we don’t have an intermediate outcome, so we set this to zero for everyone. * H20: A matrix, with each column containing data for a main-effects term in the second-stage regression (analogous to H10). * H21: Variables that interact with second-stage treatment A2 in the second-stage regression (candidate variables for deeper tailoring). * A2: Indicator for second-stage treatment * Y2: End-of-study outcome * S: Indicator for whether an individual was re-randomized (1 = re-randomized; 0 = otherwise) * c1: Contrast matrix for first-stage regression (see above) |

The function qlaci() maximizes the expected outcome for each stage of treatment given a set of moderators.

attach(dat\_adhd\_c) # with attach we can be lazy and refer to variables in the data.frame by name directly  
  
q1 <- qlaci::qlaci(H10 = cbind(1, odd\_c, severity\_c, race\_c, priormed),  
 H11 = cbind(A1 = 1, "A1:priormed" = priormed),  
 A1 = A1,  
 Y1 = rep(0, nrow(dat\_adhd\_c)), # set to zero for everyone; we care only about EOS outcome  
 H20 = cbind(1, odd\_c, severity\_c, race\_c, priormed, A1, adherence),  
 H21 = cbind(A2 = 1, "A2:A1" = A1, "A2:adherence" = adherence),  
 A2 = A2,  
 Y2 = Y2,  
 S = 1 - R,  
 c1 = t(c1))  
  
detach(dat\_adhd\_c)

## qlaci results

The the coefficients estimated by qlaci() combined with the user specified contrast matrix give us the estimated means for first-stage treatment options by levels of priormed, accounting for the optimal second-stage tactic for non-responders. With valid confidence intervals we can test contrasts and even compare to the 4 embedded adaptive interventions found in Virtual Module 2!

data.frame(q1$ci1)

est low upp  
Mean Y under bmod, prior med 3.373 2.764 3.894  
Mean Y under med, prior med 3.798 3.247 4.376  
Mean diff (bmod-med) for prior med -0.425 -1.139 0.341  
Mean Y under bmod, no prior med 4.316 3.875 4.675  
Mean Y under med, no prior med 2.927 2.512 3.328  
Mean diff (bmod-med) for no prior med 1.389 0.784 1.887  
Mean Y Optimal AI 4.161 3.799 4.458

\*Optimal AI SE not accounting for variability in estimating

### Knowledge check #4

|  |
| --- |
| Questions |
| 1. Looking at the above table, what do the two contrasts (bmod - med) for levels of prior med tell us about tailoring? 2. ? |